

# CSCE 2014 – Programming Foundations II

## Class Syllabus – Summer 2017

### General Information

Instructor: Jon C. Hammer  
Instructor's Office: JBHT 434  
Class Hours: Monday - Thursday 12:45 PM - 2:15 PM  
Instructor's Office Hours: Tuesday, Thursday 2:15 PM – 3:45 PM  
Instructor's Email: [jhammer@uark.edu](mailto:jhammer@uark.edu)  
Class Web Page: <http://moodle.csce.uark.edu>

### Catalog Listing

This course continues developing problem solving techniques by focusing on fundamental data structures and associated algorithms. Topics include: abstract data types, introduction to object-oriented programming, recursion, searching and sorting algorithms, templates, linked lists, stacks, queues, binary trees, heaps, and hash tables. Using C++ in a UNIX environment. Prerequisite: CSCE 2004.

### Text Book (Optional)

- ◆ Data Abstraction & Problem Solving with C++: Walls and Mirrors (7th Edition), Frank Carrano and Timothy Henry, 2016. Pearson.
- ◆ Previous editions of this book are acceptable and widely available.

### Grading

Final grades in this class will be determined by a weighted average of daily programming exercises, programming project scores, and exam scores as follows:

- ◆ Daily Programming Exercises: 10%
- ◆ Programming Projects: 40%
- ◆ Midterm Exam: 20%
- ◆ Final Exam: 30%

We use the following scale to assign final grades:

- ◆ **A:** over 90%
- ◆ **B:** 80% - 89%
- ◆ **C:** 70% - 79%
- ◆ **D:** 60% - 69%
- ◆ **F:** below 60%

Students must pass **EACH** individual portion of the class (daily exercises, programming projects, and exams) with a grade of **D** or better in order to pass this course. Hence, an overall average greater than 60% may still result in a failure in some cases.

## Daily Programming Exercises

Most days after the lecture, a short coding prompt will be provided by the instructor. Students are expected to complete the prompt in class and demonstrate their work to the instructor in order to receive credit for the exercise. Students are required to attend the lecture in order to receive any credit for that day's exercise.

## Programming Projects

There also will be 4-5 relatively large programming projects that will integrate material taught in the course. The project requirements and due dates will be posted on the class website. The programming projects will be graded according to the following scale:

- ◆ 40% Program Correctness
- ◆ 20% Software Design
- ◆ 10% Programming Style
- ◆ 10% Testing
- ◆ 10% Documentation
- ◆ 10% Midpoint Submission

**Midpoint and Final Submissions:** Each programming project will have **two** associated deadlines. The first will be for a **midpoint submission**. The purpose of the midpoint submission is to ensure students start working on the projects early, rather than waiting until the last day. Students are expected to submit evidence they have started working on the project to the class website in the form of current source code (.h and .cpp files). The midpoint submission **does not** need to be complete, but it **does** need to compile without errors or warnings.

The second deadline is for the **final submission**. The final submission should reflect the students' best efforts at solving the given problem and will include:

- ◆ **All code files required for compilation.** Source code should have either the .h or .cpp file extension (as appropriate). All code **must** compile without warnings and follow the Style Guide provided on the course website.
- ◆ **A single documentation file (with the .pdf file extension).** Please use the Documentation Template provided on the course website as the basis for your documentation. Make sure to answer all questions completely.
- ◆ **Evidence of testing.** This usually consists of example program input and output on normal and extraneous test cases. Testing can be supplied in another file or included as part of the documentation.

Both midpoint and final submissions for programming projects must be submitted electronically by **midnight** of the due date specified in the project description. **Projects that are late will NOT be accepted.** Partial credit **will** be given for programs that compile but are not complete. Starting early on programming projects is **strongly** encouraged.

## Exams

There will be two exams in this class, **one midterm exam** and **one comprehensive final exam**. All exams will be closed book, but **each student will be allowed to bring in a single 8.5 by 11 sheet of paper (double-sided) with any notes the student wishes to include**. Calculators will **not** be needed or allowed. Make up exams will only be allowed under exceptional circumstances (e.g., a note from your doctor).

## Academic Misconduct

The department, college, and university have very strict guidelines regarding academic misconduct. Students are expected to follow all policies specified in the University of Arkansas Undergraduate Studies Catalog and on the Academic Initiatives and Integrity website ([honesty.uark.edu](http://honesty.uark.edu)). In addition, the following policies will apply to this class.

### Exams:

- ◆ Students are **NOT** allowed to copy anything from another student.
- ◆ Students are **NOT** allowed to get any outside assistance during the exam.
- ◆ Students **ARE** allowed to bring one 8.5 x 11 sheet of paper with any notes they want into the exam. Otherwise, exams are closed book and closed notes.

### Programming Projects:

- ◆ Students **ARE** expected to submit their own work on programming projects.
- ◆ Students are **NOT** allowed to work in groups to implement programs.
- ◆ Students are **NOT** allowed to lend or distribute code to each other.
- ◆ Students are **NOT** allowed to copy code from other individuals or websites.
- ◆ Students **ARE** allowed to ask the instructor for assistance.
- ◆ Students **ARE** allowed to borrow and adapt code from this class website or the textbook.

Instances of academic misconduct will be reported to the Academic Initiatives and Integrity office, and may result in a **zero** on the project/exam, **failure** in the class, or **suspension** from the university depending on the severity of the infraction and prior violations.

## ADA Statement

If any member of the class has a documented disability and needs special accommodations, the instructor will work with the student to provide reasonable accommodation to ensure the student a fair opportunity to perform in this class. Please advise the instructor of the disability and the desired accommodations within the first week of the semester.

## Inclement Weather

If the university is officially closed, class will not be held. When the university is open, you are expected to make a reasonable effort to attend class, but not if you do not feel that you can get to campus safely. Assignment due dates will be postponed in case of inclement weather.

## **Emergency Procedures**

Many types of emergencies can occur on campus; instructions for specific emergencies such as severe weather, active shooter, or fire can be found at [emergency.uark.edu](http://emergency.uark.edu).

### **Severe Weather (Tornado Warning):**

- ◆ Follow the directions of the instructor or emergency personnel.
- ◆ Seek shelter in the basement or interior room or hallway on the lowest floor, putting as many walls as possible between you and the outside.
- ◆ If you are in a multi-story building, and you cannot get to the lowest floor, pick a hallway in the center of the building.
- ◆ Stay in the center of the room, away from exterior walls, windows, and doors.

### **Violence / Active Shooter:**

- ◆ CALL - 9-1-1
- ◆ AVOID - If possible, self-evacuate to a safe area outside the building. Follow directions of police officers.
- ◆ DENY - Barricade the door with desk, chairs, bookcases or any items. Move to a place inside the room where you are not visible. Turn off the lights and remain quiet. Remain there until told by police it is safe.
- ◆ DEFEND - Use chairs, desks, cell phones or whatever is immediately available to distract and/or defend yourself and others from attack.